INNOVATIVE TOOLS FOR SCIENTIFIC COMPUTATION
IN AERONAUTICAL ENGINEERING
J. Periaux, P. Joly, O. Pironneau and E. Oñate (Eds.)

MESHLESS ANALYSIS OF INCOMPRESSIBLE FLOWS USING THE FINITE POINT METHOD

E. Oñate, C. Sacco
International Center for Numerical Methods in Engineering (CIMNE)
Universidad Politécnica de Cataluña
Gran Capitán s/n, 08034 Barcelona, Spain
E-mail: onate@cimne.upc.es
Web page: http://www.cimne.upc.es

S. Idelsohn
Universidad Nacional del Litoral
Santa Fe, Argentina
E-mail: rnsergio@alpha.arcride.edu.ar

Abstract. A stabilized finite point method (FPM) for the meshless analysis of incompressible fluid flow problems is presented. The stabilization approach is based in the finite calculus (FIC) procedure. An enhanced fractional step procedure allowing the semi-implicit numerical solution of incompressible fluids using the FPM is described. Examples of application of the stabilized FPM to the solution of incompressible flow problems are presented.

Key words: Meshless method, finite point method, incompressible flow.

1 INTRODUCTION

Mesh free techniques have become quite popular in computational mechanics. A family of mesh free methods is based on smooth particle hydrodynamic procedures [1,2]. These techniques, also called free lagrangian methods, are typically used for problems involving large motions of solids and moving free surfaces in fluids. A second class of mesh free methods derive from generalized finite difference (GFD) techniques [3,4]. Here the approximation around each point is typically defined in terms of Taylor series expansions and the discrete equations are found by using point collocation. Among a third class of mesh free techniques we find the so called diffuse element (DE) method [5], the element free Galerkin (EFG) method [6,7] and the reproducing kernel particle (RKP) method [8,9]. These three methods use local interpolations for defining the approximate field around a point in terms of values in adjacent points, whereas the discretized system of equations is typically obtained by integrating the Galerkin variational form over a suitable background grid.

The *finite point method* (FPM) proposed in [10–15] is a truly meshless procedure. The approximation around each point is obtained by using standard moving least square techniques similarly as in DE and EFG methods. The discrete system of equations is obtained by sampling the governing differential equations at each point as in GFD methods.

The basis of the success of the FPM for solid and fluid mechanics applications is the *stabilization* of the discrete differential equations. The stable form found by the *finite calculus* procedure presented in [16–21] corrects the errors introduced by the point collocation procedure, mainly next to the boundary segments. In addition, it introduces the necessary stabilization for treating high convection effects and it also allows equal order velocity-pressure interpolations in fluid flow problems [19,21].

The content of the chapter is structured as follows. In the next section the basis of the FPM approximation is described. The stabilized governing equations for incompressible flows derived using the finite calculus (FIC) approach are then presented. Next a three step semi-implicit fractional solution scheme using the FPM approximation is described in some detail. Finally, examples of the efficiency and accuracy of the stabilized FPM for numerical solution of incompressible flow problems are presented, namely the analysis of a driven cavity flow, the solution of a backwards facing step, the analysis of a submerged cylinder and the aerodynamic study of a NACA airfoil.

2 INTERPOLATION IN THE FPM

Let Ω_i be the interpolation domain (cloud) of a function $u(x)$ and let s_j with $j = 1, 2, \cdots, n$ be a collection of n points with coordinates $x_j \in \Omega_i$. The unknown function u may be approximated within Ω_i by

$$u(x) \cong \hat{u}(x) = \sum_{l=1}^{m} p_l(x)\alpha_l = \mathbf{p}(x)^T \boldsymbol{\alpha} \tag{1}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots \alpha_m]^T$ and vector $\mathbf{p}(x)$ contains typically monomials, hereafter termed "base interpolating functions", in the space coordinates ensuring that the basis is complete. For a 2D problem we can specify

$$\mathbf{p} = [1, x, y]^T \quad \text{for} \quad m = 3 \tag{2}$$

and

$$\mathbf{p} = [1, x, y, x^2, xy, y^2]^T \quad \text{for} \quad m = 6 \qquad \text{etc.} \tag{3}$$

Function $u(x)$ can now be sampled at the n points belonging to Ω_i giving

$$\mathbf{u}^h = \begin{Bmatrix} u_1^h \\ u_2^h \\ \vdots \\ u_n^h \end{Bmatrix} \cong \begin{Bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \vdots \\ \hat{u}_n \end{Bmatrix} = \begin{Bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_n^T \end{Bmatrix} \boldsymbol{\alpha} = \mathbf{C}\boldsymbol{\alpha} \tag{4}$$

where $u_j^h = u(x_j)$ are the unknown but sought for values of function u at point j, $\hat{u}_j = \hat{u}(x_j)$ are the approximate values, and $\mathbf{p}_j = \mathbf{p}(x_j)$.

In the FE approximation the number of points is chosen so that $m = n$. In this case $\mathbf{C}$ is a square matrix. The procedure leads to the standard shape functions in the FEM [22].

If $n > m$, $\mathbf{C}$ is no longer a square matrix and the approximation can not fit all the u_j^h values. This problem can be simply overcome by determining the $\hat{u}$ values by minimizing the sum of the square distances of the error at each point weighted with a function $\varphi(x)$ as

$$J = \sum_{j=1}^{n} \varphi(x_j)(u_j^h - \hat{u}(x_j))^2 = \sum_{j=1}^{n} \varphi(x_j)(u_j^h - \mathbf{p}_j^T \boldsymbol{\alpha})^2 \tag{5}$$

with respect to the $\boldsymbol{\alpha}$ parameters. Note that for $\varphi(x) = 1$ the standard least square (LSQ) method is reproduced.

Function $\varphi(x)$ is usually built in such a way that it takes a unit value in the vecinity of the point i typically called "star node" where the function (or its derivatives) are to be computed and vanishes outside a region Ω_i surrounding the point. The region Ω_i can be used to define the number of sampling points n in the interpolation region. A typical choice for $\varphi(x)$ is the normalized Gaussian function and this has been chosen in the examples shown in the paper. Of course $n \geq m$ is always required in the sampling region and if equality occurs no effect of weighting is present and the interpolation is the same as in the LSQ scheme.

Standard minimization of eq.(5) with respect to $\boldsymbol{\alpha}$ gives

$$\boldsymbol{\alpha} = \bar{\mathbf{C}}^{-1}\mathbf{u}^h, \qquad \bar{\mathbf{C}}^{-1} = \mathbf{A}^{-1}\mathbf{B} \tag{6}$$

$$\mathbf{A} = \sum_{j=1}^{n} \varphi(x_j)\mathbf{p}(x_j)\mathbf{p}^T(x_j) \tag{7}$$

$$\mathbf{B} = [\varphi(x_1)\mathbf{p}(x_1), \varphi(x_2)\mathbf{p}(x_2), \cdot\varphi(x_n)\mathbf{p}(x_n)]$$

The final approximation is obtained by substituting $\boldsymbol{\alpha}$ from eq.(6) into (1) giving

$$\hat{u}(x) = \mathbf{p}^T\bar{\mathbf{C}}^{-1}\mathbf{u}^h = \mathbf{N}^T\mathbf{u}^h = \sum_{j=1}^{n} N_j^i u_j^h \tag{8}$$

where the "shape functions" for the i-th star node are

$$N_j^i(x) = \sum_{l=1}^{m} p_l(x)\bar{C}_{lj}^{-1} = \mathbf{p}^T(x)\bar{\mathbf{C}}^{-1} \tag{9}$$

It must be noted that accordingly to the least square character of the approximation

$$u(x_j) \simeq \hat{u}(x_j) \neq u_j^h \tag{10}$$

i.e. the local values of the approximating function do not fit the nodal unknown values. Indeed $\hat{u}$ is the true approximation for which we shall seek the satisfaction of the differential equation and the boundary conditions and u_j^h are simply the unknown parameters sought.

The weighted least square approximation described above depends on a great extend on the shape and the way to apply the weighting function. The simplest way is to define a fixed function $\varphi(x)$ for each of the Ω_i interpolation domains [11,12].

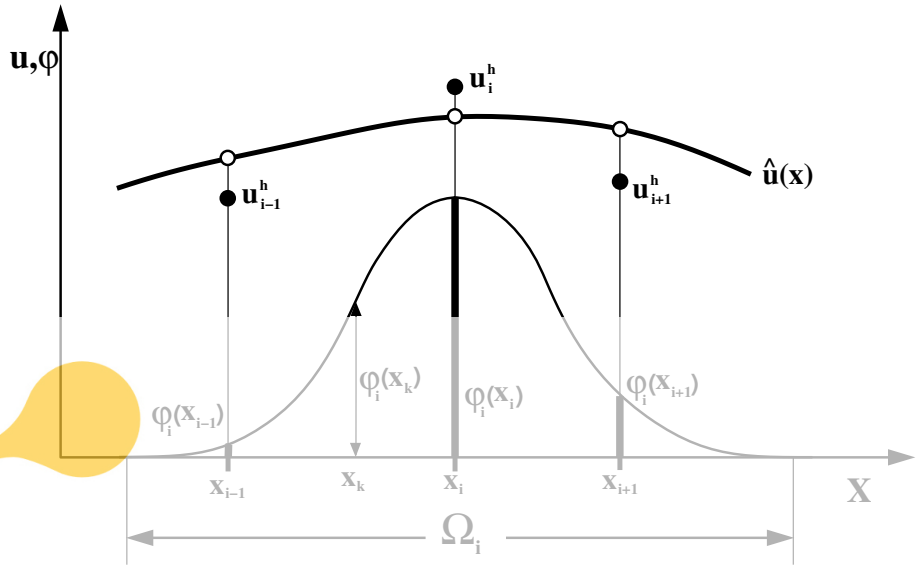


Figure 1. Fixed weighting least square procedure

Let $\varphi_i(x)$ be a weighting functions satisfying (Figure 1)

$$\begin{aligned} \varphi_i(x_i) &= 1 \\ \varphi_i(x) &\neq 0 \qquad x \in \Omega_i \\ \varphi_i(x) &= 0 \qquad x \notin \Omega_i \end{aligned} \tag{11}$$

Then the minimization square distance becomes

$$J_i = \sum_{j=1}^{n} \varphi_i(x_j)(u_j^h - \hat{u}(x_j))^2 \quad \text{minimum} \tag{12}$$

The expression of matrices **A** and **B** coincide with eq.(7) with $\varphi(x_j) = \varphi_i(x_j)$.

Note that according to (1), the approximate function $\hat{u}(x)$ is defined in each interpolation domain Ω_i. In fact, different interpolation domains can yield different shape functions N_j^i. As a consequence a point belonging to two or more overlapping interpolation domains has different values of the shape functions which means that $N_j^i \neq N_j^k$. The interpolation is now multivalued within Ω_i and, therefore for any useful approximation a decision must be taken limiting the choice to a single value. Indeed, the approximate function $\hat{u}(x)$ will be typically used to provide the value of the unknown function $u(x)$ and its derivatives in only specific regions within each interpolation domain. For instance by using point collocation we may limit the validity of the interpolation to a single point x_i. It is precisely in this context where we have found this meshless method to be more useful for practical purposes [10–15].

3 STABILIZED FPM USING A FINITE CALCULUS APPROACH

Finite element solution of the incompressible Navier-Stokes equations with the classical Galerkin method may suffer from numerical instabilities from two main sources. The first is due to the advective-diffusive character of the equations which induces oscillations for high values of the velocity. The second source has to do with the mixed character of the equations which limits the choice of finite element interpolations for the velocity and pressure fields.

Solutions of these two problems have been extensively sought in the last years. Compatible velocity-pressure interpolations satisfying the inf-sup condition emanating from the second problem above mentioned have been used. In addition, the advective operator has been modified to include some "upwinding" effects [22–30]. Recent procedures based on Galerkin Least Square [31,32], Characteristic Galerkin [33,34], Variational Multiscale [35–37] and Residual Free Bubbles [38–40] techniques allow equal order interpolation for velocity and pressure by introducing a Laplacian of pressure term in the mass balance equation, while preserving the upwinding stabilization of the momentum equations. Most of these methods lack enough stability in the presence of sharp layers transversal to the velocity. This defficiency is usually corrected by adding new "shock capturing" stabilization terms to the already stabilized equations [41–43]. The computation of the stabilization parameters in all these methods is based in "ad hoc" generalizations of the parameters for the 1D linear advective-diffusive-reactive problem [44,45].

This paper presents a different point view for deriving stabilized a finite point method for incompressible flow problems. The starting point are the stabilized form of the governing differential equations derived via a *finite calculus* (FIC) procedure. This technique first presented in [16,17] is based on writting the different balance equations over a domain of finite size and retaining higher order terms. These terms incorporate the ingredients for the necessary stabilization of any transient and steady state numerical solution *already at the differential equations level*. Application of the MLS interpolation and point collocation to the consistently modified differential equations for the fluid flow problem leads to a stabilized system of discretized equations which overcomes *the two problems* above mentioned, i.e. the advective type instability and that due to lack of compatibility between the velocity and pressure fields.

For the sake of preciseness the basic ideas of the FIC method are given next.

3.1 Basic concept of the finite increment calculus (FIC) method

Let us consider a sourceless transient problem over a one dimensional domain AB of length L (Figure 2). The balance of flux q over a domain of finite size belonging to L can be written as

$$q_A - q_B = 0 \tag{13}$$

where A and B are the end points of the finite size domain of length h. As usual q_A and q_B represent the values of the flux q at points A and B, respectively.

For instance, in an 1D advective-diffusive problem the flux $q = -cu\phi + k\frac{d\phi}{dx}$, where ϕ is the transported variable (i.e. the temperature in a thermal problem),

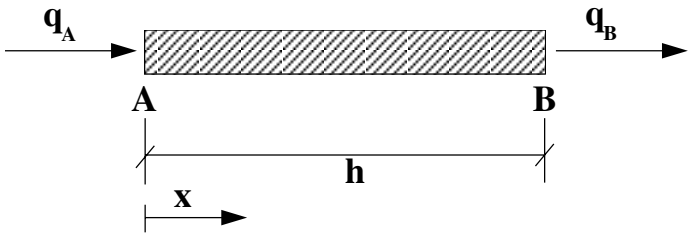


Figure 2. Equilibrium of fluxes in a finite balance domain

u is the advective velocity and c and k are the advective and diffusive material parameters, respectively.

The flux q_A can be expressed in terms of the values at point B by the following Taylor series expansion

$$q_A = q_B - h\frac{\partial q}{\partial x}|_B + \frac{h^2}{2}\frac{d^2q}{dx^2}|_B + Oh^3 \tag{14}$$

Substituting (14) into (13) gives after simplification and neglecting cubic terms in h

$$\frac{dq}{dx} - \underline{\frac{h}{2}\frac{dq}{dx}} = 0 \tag{15}$$

where all terms are evaluated at the arbitrary point B.

Eq. (15) is the *finite* form of the balance equation over the domain AB. The underlined term in eq.(15) introduces the necessary stabilization for the discrete solution of eq.(15) using *any* numerical technique. Distance h is the characteristic length of the discrete problem and its value depends on the parameters of discretization method chosen (such as the grid size). Note that for $h \to 0$ the standard infinitesimal form of the balance equation $\left(\frac{dq}{dx} = 0\right)$ is recovered.

Above process can be extended to derive the stabilized balance differencial equations for any problem in mechanics as

$$r_d - \underline{\frac{h_j}{2}\frac{\partial r_i}{\partial x_j}} = 0 \tag{16}$$

where r_i is the standard form of the ith differential equation for the infinitesimal problem, h_j are the dimensions of the domain where balance of fluxes, forces, etc. is enforced, and $j = 1, 2, 3$ for 3D problems. Details of the derivation of eq.(16) for steady-state and transient advective-diffusive and fluid flow problems can be found in [16]. Applications of the FIC approach to the Galerkin finite element solution of these problems are reported in [16–21].

The underlined stabilization terms in eqs.(15) and (16) are a consequence of accepting that the infinitesimal form of the balance equations is an unreachable limit within the framework of a discrete numerical solution. Indeed eqs.(3) or (4) are *not longer valid* for obtaining an analytical solution following traditional integration methods from infinitesimal calculus theory. The meaning of the new

stabilized equations makes only sense in the context of a discrete numerical method yielding approximate values of the solution at a finite set of points within the analysis domain. Convergence to the *exact* analytical value at the points will occur only for the limit case of zero grid size (except for some simple 1D problems [16]) which also implies naturally a zero value of the characteristic length parameters.

The FIC formulation presented below for incompressible flows can be considered an extension of that recently developed in [21] for finite element analysis of incompressible Navier-Stokes flows. The set of stabilized governing equations is first discretized in time using a semi-implicit fractional step procedure and then solved in space using the FPM. The stabilized formulation allows the use of an equal order interpolation for the velocities and pressure variables.

3.2 FIC formulation of viscous flow equations

We consider the motion around a body of a viscous incompressible fluid.

The stabilized FIC form of the governing differential equations for the three dimensional (3D) problem can be written as

Momentum

$$r_{m_i} - \underline{\frac{1}{2}h_{mj}\frac{\partial r_{m_i}}{\partial x_j}} - \underline{\frac{1}{2}\delta\frac{\partial r_{m_i}}{\partial t}} = 0 \qquad \text{on } \Omega \quad i,j = 1,2,3 \tag{17}$$

Mass balance

$$r_d + \underline{\frac{1}{2}h_{dj}\frac{\partial r_d}{\partial x_j}} = 0 \qquad \text{on } \Omega \quad j = 1,2,3 \tag{18}$$

where

$$r_{m_i} = \rho\left[\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j}(u_i u_j)\right] + \frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} - b_i \tag{19}$$

$$r_d = \frac{\partial u_i}{\partial x_i} \qquad i = 1,2,3 \tag{20}$$

In above u_i is the velocity along the i-th global reference axis, ρ is the (constant) density of the fluid, p is the pressure, b_i are the body forces acting in the fluid and τ_{ij} are the viscous stresses related to the viscosity μ by the standard expression

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \delta_{ij}\frac{2}{3}\frac{\partial u_k}{\partial x_k}\right) \tag{21}$$

The boundary conditions for the stabilized problem are written as

$$n_j\tau_{ij} + t_i + \underline{\frac{1}{2}h_{mj}n_j r_{m_i}} = 0 \qquad \text{on } \Gamma_t \tag{22}$$

$$u_j - u_j^p = 0 \qquad \text{on } \Gamma_u \tag{23}$$

where n_j are the components of the unit normal vector to the boundary and t_i and u_j^p are prescribed tractions and displacements on the boundaries Γ_t and Γ_u, respectively.

The underlined terms in eqs.(17)–(22) introduce the necessary stabilization for the approximated numerical solution.

The *characteristic length* distances h_{mj} and h_{dj} represent the dimensions of the finite domain where balance of momentum and mass. The signs before the stabilization terms in eqs.(17), (19) and (22) ensure a positive value of the characteristic length distances. The parameter δ in eq.(17) has dimensions of time. Details of the derivation of eqs. (17)–(23) can be found in [16,19,21].

Eqs.(17–23) are the starting point for deriving a variety of stabilized numerical methods for solving the incompressible Navier-Stokes equations. It can be shown that a number of standard stabilized finite element methods allowing equal order interpolations for the velocity and pressure fields can be recovered from the modified form of the momentum and mass balance equations given above [16,19].

Alternative form of the mass balance equation

Taking the first derivative of eq.(21) gives (assuming the viscosity μ to be constant)

$$\frac{\partial \tau_{ij}}{\partial x_j} = \mu \Delta u_i + \frac{\mu}{3} \frac{\partial r_d}{\partial x_i} \tag{24}$$

where $\Delta = \frac{\partial^2}{\partial x_i \partial x_i}$ is the Laplacian operator. Substituting eq.(24) into (17) gives after small algebra

$$\frac{\partial r_d}{\partial x_i} \left(\frac{\mu}{3} + \frac{\rho u_i h_{m_i}}{2} \right)^{-1} \left[r_{m_i} - \frac{h_{m_k}}{2} \frac{\partial r_{m_i}}{\partial x_k} + \frac{\rho u_i h_{m_i}}{2} \frac{\partial r_d}{\partial x_i} - \frac{\delta}{2} \frac{\partial r_{m_i}}{\partial t} \right] \quad \text{no sum in } i \tag{25}$$

where

$$\bar{r}_{m_i} = r_{m_i} + \frac{\mu}{3} \frac{\partial r_d}{\partial x_i} \tag{26}$$

and r_{m_i} is given by eq.(19).

Inserting eq.(25) into eq.(18) gives

$$r_d + c_i \left(\bar{r}_{m_i} - \frac{h_{m_k}}{2} \frac{\partial r_{m_i}}{\partial x_k} + \frac{\rho u_i h_{m_i}}{2} \frac{\partial r_d}{\partial x_i} - \frac{\delta}{2} \frac{\partial r_{m_i}}{\partial t} \right) = 0 \quad \text{no sum in } i \tag{27}$$

with

$$c_i = \left(\frac{2\mu}{3 h_{d_i}} + \frac{\rho u_i h_{m_i}}{h_{d_i}} \right)^{-1} \qquad \text{no sum in } i \tag{28}$$

Eq.(27) can be rewritten as

$$r_d - g_{ii} \frac{\partial^2 p}{\partial x_i \partial x_i} + r_p = 0 \tag{29}$$

where

$$r_p = c_i \bar{r}_{m_i} - g_{ij} \frac{\partial}{\partial x_j} \left(r_{m_i} - \delta_{ij} \frac{\partial p}{\partial x_i} \right) + \frac{\rho u_i h_{m_i}}{2} \frac{\partial r_d}{\partial x_i} - \frac{\delta}{2} \frac{\partial r_{m_i}}{\partial t} \quad \text{no sum in } i \tag{30}$$

and

$$g_{ij} = \left(\frac{4\mu}{3 h_{d_i} h_{m_j}} + \frac{2\rho u_i h_{m_i}}{h_{d_i} h_{m_j}} \right)^{-1} \quad \text{no sum in } i \tag{31}$$

Note that for $h_{m_i} = h_{m_j} = h$ where h is a typical grid dimension (i.e. the average size of a cloud of points), the value of g_{ii} is simply

$$g_{ii} = \left(\frac{4\mu}{3h^2} + \frac{2\rho u_i}{h} \right)^{-1} \tag{32}$$

The stabilization parameter g_{ii} has now the form traditionally used in the Galerkin Least Square formulation for the viscous (Stokes) limit ($u_i = 0$) and the inviscid (Euler) limit ($\mu = 0$) and deduced from ad-hoc extensions of the 1D advective-diffusive problem [25–46]. Note, however, that the general form of the stabilization parameter g_{ii} is deduced here from the general FIC formulation without further extrinsic assumptions.

Indeed, the precise computation of the characteristic length values is crucial for the practical applications of above stabilized expressions. This topic is dealt with on Section 7.

4 FRACTIONAL STEP APPROACH

The momentum equations (17) are first discretized in time using the following scheme

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\rho} \left[\rho \frac{\partial (u_i u_j)^n}{\partial x_j} + \frac{\partial p^{n+1}}{\partial x_i} - \frac{\partial \tau_{ij}^n}{\partial x_j} - b_i^n - \frac{h_{m_k}^n}{2} \frac{\partial r_{m_i}^n}{\partial x_k} - \frac{\delta^n}{2} \frac{\partial r_{m_i}^n}{\partial t} \right] \tag{33}$$

Eq.(33) is now split into the two following equations

$$u_i^* = u_i^n - \frac{\Delta t}{\rho} \left[\rho \frac{\partial (u_i u_j)}{\partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} - b_i - \frac{h_{m_k}}{2} \frac{\partial r_{m_i}}{\partial x_k} - \frac{\delta}{2} \frac{\partial r_{m_i}}{\partial t} \right]^n \tag{34}$$

$$u_i^{n+1} = u_i^* - \frac{\Delta t}{\rho} \frac{\partial p^{n+1}}{\partial x_i} \tag{35}$$

Note that the sum of eqs.(34) and (35) gives the original form of eq.(33).

Substituting eq.(35) into the stabilized mass balance equation (29) gives the standard Laplacian of pressure form

$$\left(\frac{\Delta t}{\rho} + g_{ii}^n \right) \frac{\partial^2 p^{n+1}}{\partial x_i \partial x_i} = r_d^* + r_p^n \tag{36a}$$

where

$$r_d^* = \frac{\partial u_i^*}{\partial x_i} \tag{36b}$$

Standard fractional step procedures neglect the contribution from the terms involving g_{ii} in eq. (36a). These terms have an additional stabilization effect which improves the numerical solution when the values of Δt are small. Note that for $\Delta t \to 0$ the term g_{ii} introduces the necessary stability in the laplacian equation, thereby overcoming the Babuska-Brezzi conditions and allowing for equal order interpolation of the velocities and pressure variables [22].

A typical solution in time includes the following steps.

Step 1. Solve explicitely for the so called fractional velocities u_i^* using eq. (33).

Step 2. Solve for the pressure field p^{n+1} solving the laplacian equation (36a).

Step 3. Compute the velocity field u_i^{n+1} for each mesh node using eq.(35)

5 NUMERICAL SOLUTION USING THE FPM

The implementation of the three step scheme described in previous section in the context of the FPM is straight forward. Eq. (8) is used to define the approximation of velocities and pressures within each cloud of point Ω_i as

$$\hat{u}_m = \sum_{j=1}^{n} N_j^i u_{m_j}^h; \quad m = 1, 2, 3 \quad \text{for 3D} \tag{39}$$

$$\hat{p} = \sum_{j=1}^{n} N_j^i p_j^h \tag{40}$$

where $(\hat{\cdot})$ denotes approximate values and the shape functions N_j^i were defined in eq.(9).

Direct substitution of eqs.(39) and (40) into the stabilized governing equations described in previous section gives the following numerical scheme for computation of the point parameters $u_{m_j}^h$ and p_j^h.

Step 1. Computation of fractional velocities

Compute *explicitely* the fractional velocities at each point k in the domain as

$$(\hat{u}_i^*)_k = (\hat{f}_i^n)_k; \quad k = 1, \ldots, N; \quad i = 1, 2, 3 \tag{41}$$

in which N is the total number of points in the domain and

$$(\hat{f}_i^n)_k = \left\{ \hat{u}_i^n - \frac{\Delta t}{\rho} \left[\rho \frac{\partial (\hat{u}_i \hat{u}_j)}{\partial x_j} - \frac{\partial \hat{\tau}_{ij}}{\partial x_j} - b_i - \frac{h_{m_j}}{2} \frac{\partial \hat{r}_{m_i}}{\partial x_j} - \frac{\delta}{2} \frac{\partial \hat{r}_{m_i}}{\partial t} \right]^n \right\}_k \tag{42}$$

where $(\hat{\cdot})$ denotes approximate values.

Once the values of $\hat{u}_i^*$ have been obtained, the parameters $u_{m_j}^h$ can be computed at each point by solving the following system of equations

$$(\hat{u}_m^*)_k = \sum_{j=1}^{n} N_j^k u_{m_j}^h, \quad k = 1, \ldots, N \tag{43}$$

Eq.(43) is a system of N equations with N unknowns from where the parameters $u_{m_j}^h, \quad j = 1, \ldots, N$ can be found. These parameters are needed to compute the derivatives of the velocity field in steps 2 and 3. Indeed the solution of eq.(43) must be repeated for every component of the velocity vector (i.e. $m = 1, 2, 3$ for 3D problems).

Step 2. Computation of pressures at time $n+1$

Compute the pressure field at time $n+1$ by solving eq.(36a). Substituting eqs. (40) and (43) into (36a) and sampling this equation at each point in the domain gives

$$\mathbf{K}^n(\mathbf{p}^h)^{n+1} = \hat{\mathbf{r}}_d^* + \hat{\mathbf{r}}_p^n \tag{44}$$

where (for 2D problems)

$$K_{kj}^n = (\frac{\Delta t}{\rho} + \hat{g}_{ii}^n) \left(\frac{\partial^2 N_j^k}{\partial x_1^2} + \frac{\partial^2 N_j^k}{\partial x_2^2} \right) \tag{45}$$

$$\begin{aligned} \hat{r}_{d_k}^* =& \hat{c}_i \hat{\bar{r}}_{m_i} - \hat{g}_{ij} \frac{\partial}{\partial x_j} \left(\hat{r}_{m_i} - \delta_{ij} \frac{\partial \hat{p}}{\partial x_i} \right) + \frac{\rho \hat{u}_i h_{m_i}}{2} \frac{\partial \hat{r}_d}{\partial x_i} - \frac{\delta}{2} \frac{\partial \hat{r}_{m_i}}{\partial t} \quad \text{no sum in } i \\ \hat{r}_{p_k}^* =& \left[\frac{\partial \hat{u}_i^*}{\partial x_i} \right] \end{aligned} \tag{46}$$

Eq.(46) provides a system of equations from which the pressure parameters $(p_k^h)^{n+1}$ can be found at each point k.

Step 3. Computation of velocities at time $n+1$

The final step is the explicit computation of the velocities in each point at time $n+1$. Substituting the known values of $\hat{u}_i$ and $\hat{p}^{n+1}$ at each point into eq.(35) gives

$$(\hat{u}_i^{n+1}) = \left[\hat{u}_i^* - \frac{\Delta t}{\rho} \frac{\partial \hat{p}^{n+1}}{\partial x_i} \right]_k ; \quad k = 1, \ldots, N \tag{47}$$

Note that the derivatives of the approximate functions $\hat{u}_i$ and $\hat{p}$ are computed by direct differentiation of the expressions (39) and (40), i.e.

$$\begin{aligned} \frac{\partial \hat{u}_m}{\partial x_l} &= \sum_{j=1}^{n} \frac{\partial N_j^i}{\partial x_l} u_{m_j}^h \\ \frac{\partial \hat{p}}{\partial x_l} &= \sum_{j=1}^{n} \frac{\partial N_j^i}{\partial x_l} p_j^h \end{aligned} \tag{48}$$

The steps 1–3 described above are repeated for every new time increment.

A local time step size for each point in the domain can be used to speed up the search of the steady state solution. The local time step is defined as $\Delta t_i = \frac{d_i}{2|\mathbf{u}_i|}$, where d_i is the minimum distance from a star point to any of its neighbourghs in the cloud. Note however that the full transient solution requires invariably the use of a global time step Δt_g equal for all nodes and defined as $\Delta t_g = \min(\Delta t_i), \quad i = 1, \ldots, N$.

6 BOUNDARY CONDITIONS

Prescribed tractions on the Neumann boundary Γ_t, (eq.(22)) or prescribed velocities at the Dirichlet boundary Γ_u (eq.(23)) may be imposed.

During the fractional step solution, the first explicit step is solved without imposing any boundary conditions. During the second step, two kinds of boundary conditions may be imposed: on boundaries where the normal velocity is imposed to the value u_n^p, eq.(23) reads using (35)

$$u_n^p = u_i^* n_i - \frac{\Delta t}{\rho} \frac{\partial p^{n+1}}{\partial x_i} n_i \tag{49}$$

Eq.(49) is a Neumann boundary condition for the pressure equation (36a). This equation is imposed in the FPM during the pressure computation (step 2) as a new equation for all points k belonging to the part of the boundary Γ_u where the normal velocity is prescribed.

On outflow boundaries with $n_j \sigma_{ij} = 0$ the pressure is imposed to a constant value, i.e. $p = 0$. In the FPM, essential boundary conditions such as $p = 0$ are imposed using the definition of the function itself via eq.(40) as

$$\hat{p}_i = \sum_{j=1}^{n} N_j^i p_j^h = 0 \tag{50}$$

Equation (50) is sampled at the points located at a boundary where $p = 0$.

During the third step the velocities are computed at all points using eq.(47) at all points within the analysis domain. In points where a velocity is imposed as an essential boundary condition, the imposed velocity value is asigned directly to the point. Next, the nodal velocity parameters $u_{m_j}^h$ are computed by solving the same system of equations described by eq.(43). For points over Neumann boundaries, in particular on boundaries where the tractions are prescribed to zero, the discretized form of eq.(22), i.e.

$$n_j \hat{\tau}_{ij} + \frac{1}{2} h_{m_j} n_j \hat{r}_{m_i} = 0 \tag{51}$$

is used for computing the velocities at the boundary points.

7 COMPUTATION OF THE STABILIZATION PARAMETERS

Accurate evaluation of the stabilization parameters is one of the crucial issues in stabilized methods. Most of existing methods use expressions which are direct extensions of the values obtained for the simplest 1D case. It is also usual to accept the so called "streamline upwind" assumption. It can be shown that this is equivalent to admit that vector $\mathbf{h}_m$ has the direction of the velocity field [16,19]. This unnecessary restriction leads to instabilities when sharp layers transversal to the velocity direction are present. This additional defficiency is usually corrected by adding a shock capturing or crosswind stabilization term [41–43]. In the FIC approach the crosswind stabilization is naturally introduced into the discretized equations through the general form of the characteristic length vector.

Let us first assume for simplicity that the stabilization parameters for the mass balance equations are the same than those for the momentum equations. This implies

$$\mathbf{h}_m = \mathbf{h}_d = \mathbf{h} \tag{52}$$

The problem remains now finding the value of the characteristic length vectors $\mathbf{h}$. Indeed, the components of $\mathbf{h}$ introduce the necessary stabilization along the streamline and transversal directions to the flow.

Excellent results have been obtained in all examples by using the same value of the characteristic length vector for each momentum equation defined by

$$\mathbf{h} = h_s \frac{\mathbf{u}}{|\mathbf{u}|} + h_c \frac{\boldsymbol{\nabla} u}{|\boldsymbol{\nabla} u|} \tag{53}$$

where $u = |\mathbf{u}|$ and h_s and h_c are the "streamline" and "cross wind" length parameters given by

$$h_s = \max(\mathbf{l}_j^T \mathbf{u})/|\mathbf{u}| \tag{54}$$

$$h_c = \max(\mathbf{l}_j^T \boldsymbol{\nabla} u)/|\boldsymbol{\nabla} u| \quad , \quad j = 1, 2, \cdots n \tag{55}$$

where $\mathbf{l}_j$ are the vectors linking each node in the cloud with the star node.

Note that the cross-wind terms in eq.(53) account for the effect of the gradient of the velocity field in the stabilization parameters. This is an standard assumption in most "shock-capturing" stabilization procedures [41–43].

Regarding the time stabilization parameter δ and in eq.(17) the value $\delta = \Delta t$ has been taken for the solution of the examples presented in the paper.

8 NUMERICAL EXAMPLES

The following examples have been solved with the FPM presented in previous section using a Gaussian weighting function in the WLS approximation and quadratic interpolation ($m = 6$) for the both the velocities and the pressure. Typically each cloud contains nine points ($n = 9$) which are chosen using a quadrant search scheme (i.e. the star node plus the two closest points within each quadrant are selected) [11-13].

8.1 Driven cavity flow at $Re = 1000$

This is a classical test problem to evaluate the behaviour of any fluid dynamic algorithm. A viscous flow is confined in a square cavity while one of its edges slides tangentially. The boundary conditions are $u = v = 0$ in 3 edges and $u = 1$, $v = 0$ on the upper edge. The problem is solved with the FPM using the distribution of 3,329 points shown in Figure 3. Initially, except at the edge, the velocity is set to zero everywhere including at the nodes located at the left and right top corners (ramp condition).

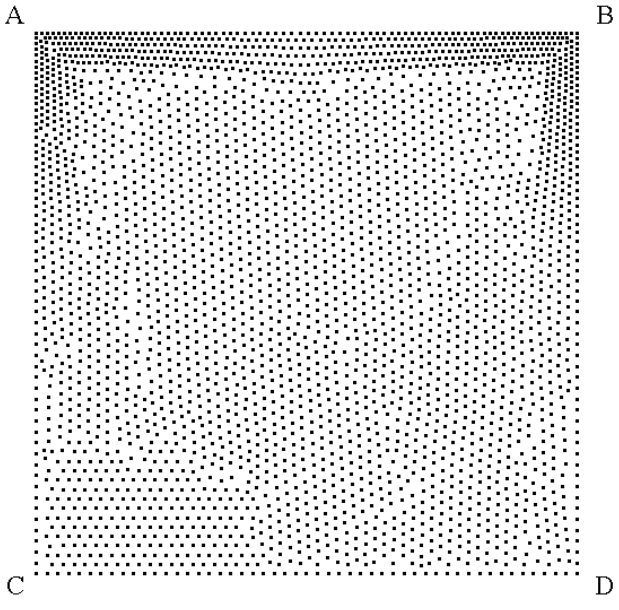


Figure 3. Driven cavity flow. Distribution of 3,329 points. Boundary conditions $\mathbf{u} = 0$ at edges AC, CD and BD and points A and B. $u = 1$ and $v = 0$ over the interior of line AB

Numerical results are shown in Figures 4, 5 and 6 for $Re = 1000$. Figures 4 and 5 show the velocity and pressure contours, respectively. The FPM results are compared with experimental results obtained by Ghia *et al.* [46] showing the velocity x computed along a vertical central cut (Figure 6). The comparison is satisfactory.

8.2 Backwards facing step at $Re = 389$

In this example, the flow is contrained to move in a 2D domain which presents a backwards step. The domain dimensions are presented in Figure 7. The step is one half the width of the inflow.

At the inflow a constant velocity profile is fixed while at the outflow the pressure is prescribed, being the velocity free. The non-slip condition is used at the walls, except for the two inflow points, where the constant inflow velocity is imposed. No volume forces are present.

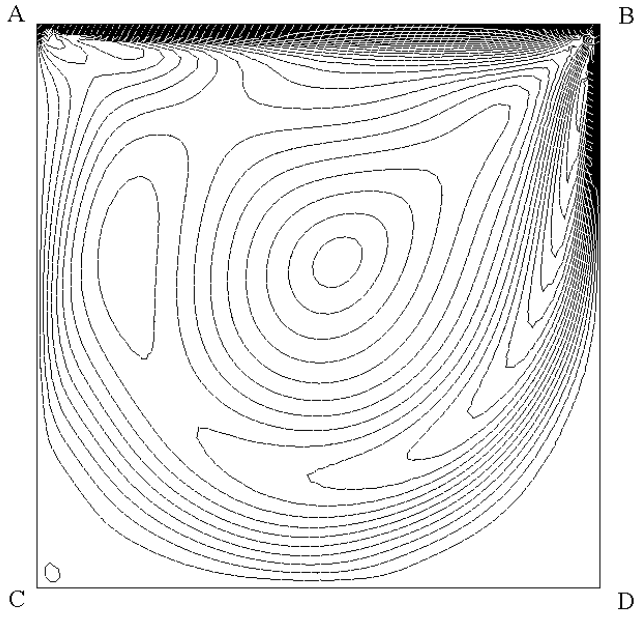


Figure 4. Driven cavity flow. Velocity contours for $Re = 1000$

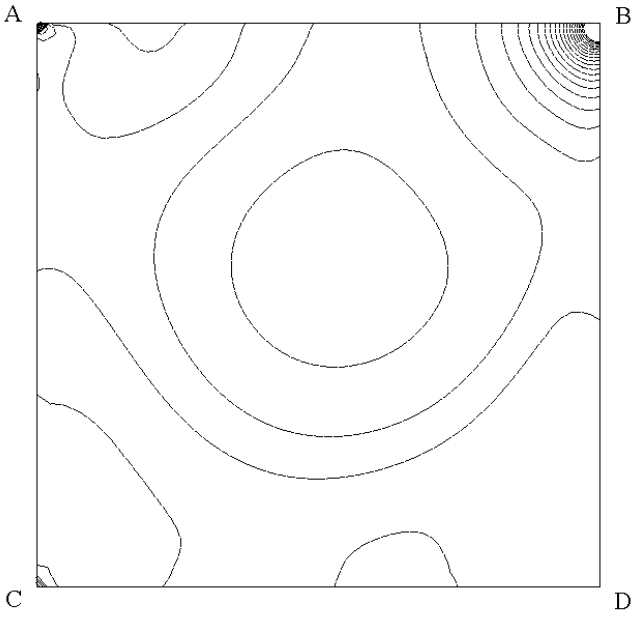


Figure 5. Driven cavity flow. Pressure contours for $Re = 1000$

The distribution of 8,462 points used near the step is represented on Figure 8. In the rest of the domain a regular distribution of points is used.

Once the stationary state is reached, the solution shows horizontal velocities represented on Figures 9 and 10 for two planes located at $x = 2.55$ S and $x = 6.11$ S from the step. The FPM results are compared with experimental results presented on ref.[47] showing an excellent agreement.

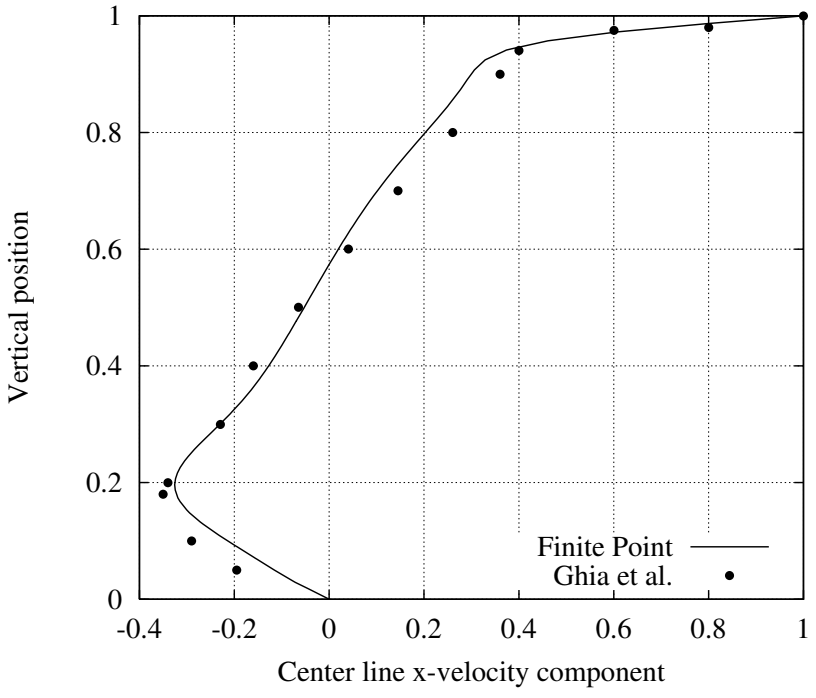


Figure 6. Driven cavity flow. Horizontal velocity distribution over the center line

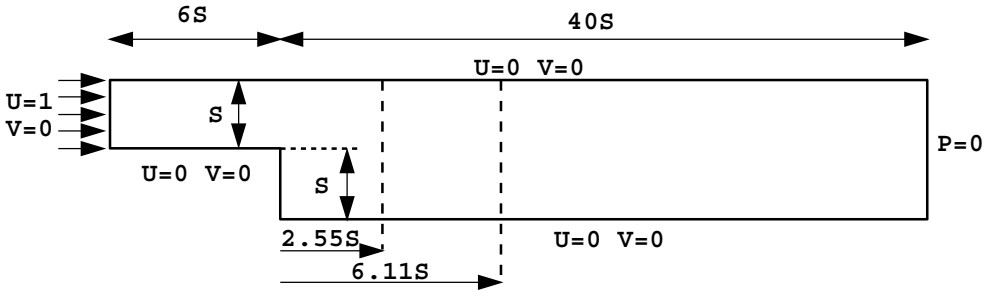


Figure 7. Backwards facing step. Geometry and boundary conditions

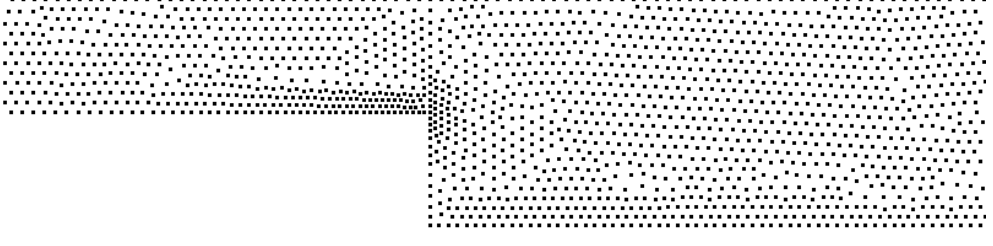


Figure 8. Backwards facing step. Distribution of 8,462 points

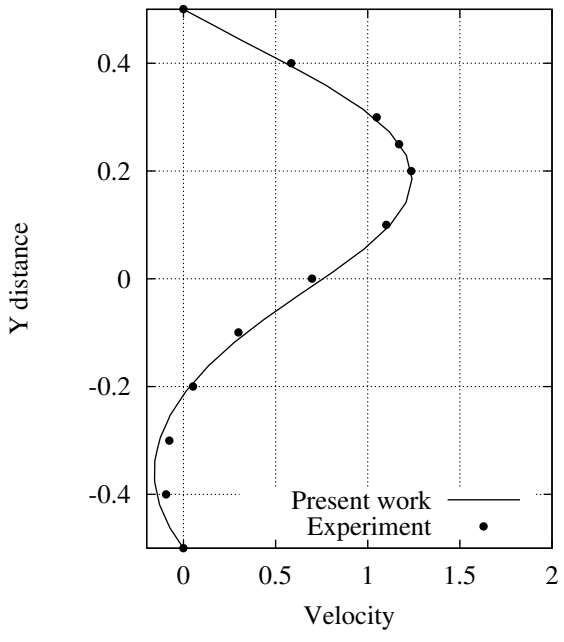


Figure 9. Backwards facing step. Horizontal velocity distribution along a vertical line at $x = 2.55$ S

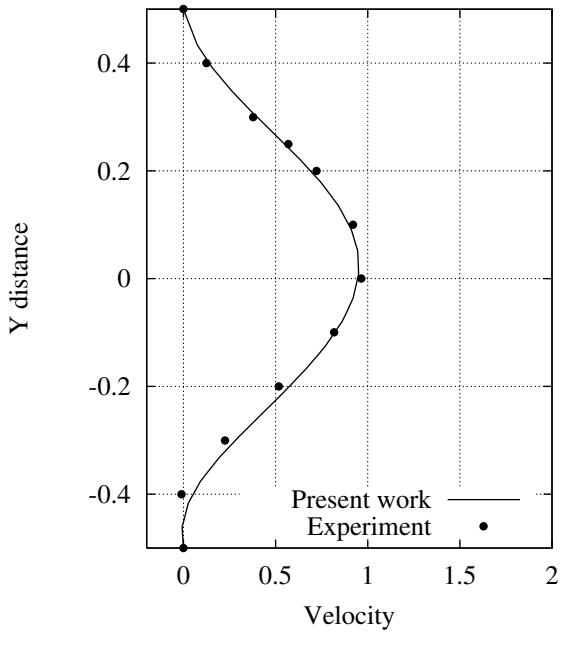


Figure 10. Backwards facing step. Horizontal velocity along a vertical line at $x = 6.11$ S

8.3 2D viscous flow around a cylinder

Figure 11 shows the geometry of the analysis domain and the boundary conditions. The problem was solved for $Re = 100$ assuming laminar flow conditions. An arbitrary grid of 9418 points was chosen for the analysis (Figure 12). The

transient analysis was run for 10000 time steps. The steady state solution was found after 18000 time steps. Note that a full period in the solution requires just 321 time steps.

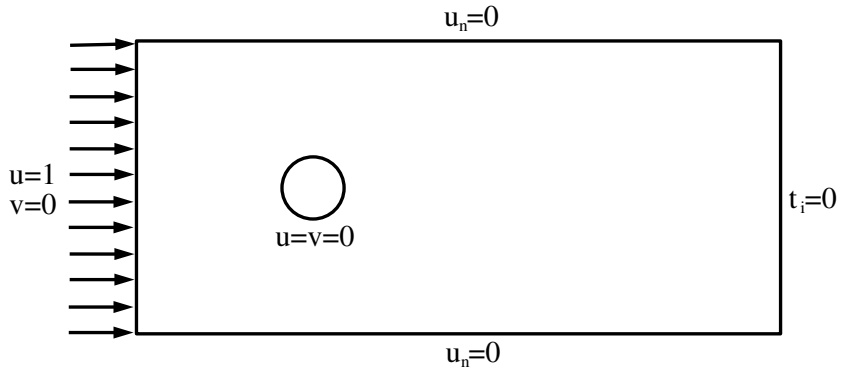


Figure 11. 2D flow around a cylinder. Analysis domain and boundary conditions. $Re = 100$. Boundary tractions (t_i) are assumed to be zero at the exit boundary

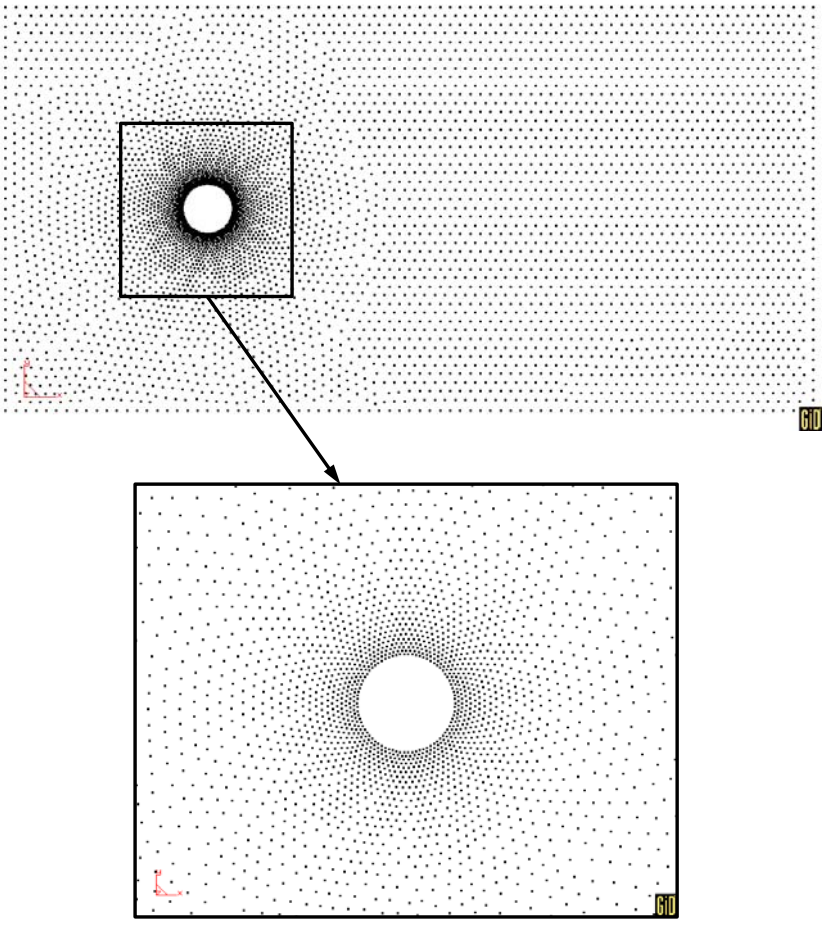


Figure 12. Grid of 9418 points used for analysis of the 2D flow around a cylinder

Figure 13 shows the velocity contour lines at four different times. Note the oscilatory character of the solution. The time evolution of the lift force is shown in Figure 14. The oscillation period deduced from the computation is 6.01 sec. This value compares well with the experimental result of 5.98 sec. ($\simeq 0.5\%$ error) reported by Roshko [48].

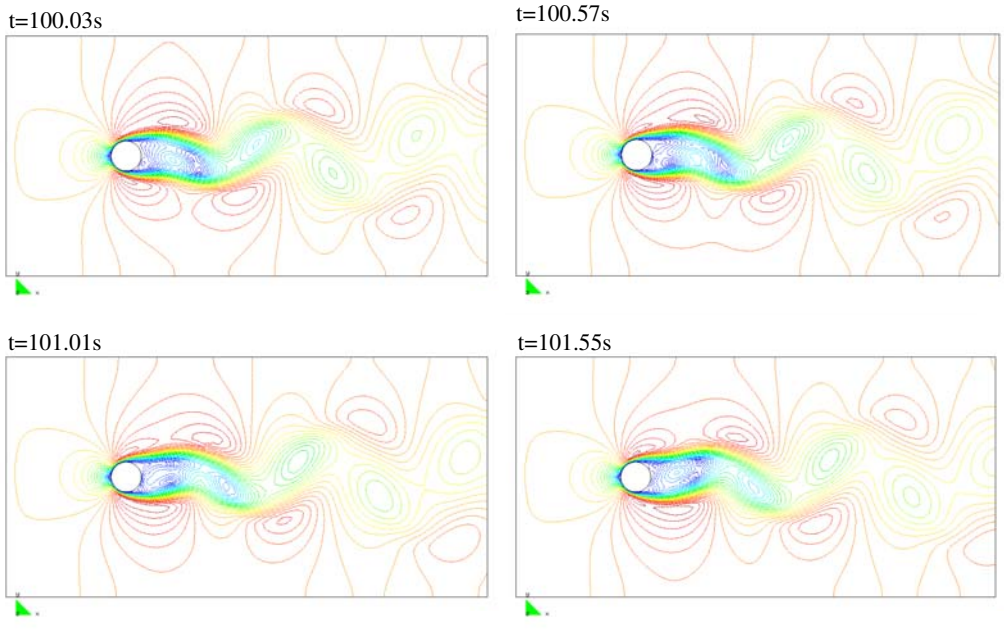


Figure 13. 2D flow around a cylinder ($Re = 100$). Velocity streamlines at different times

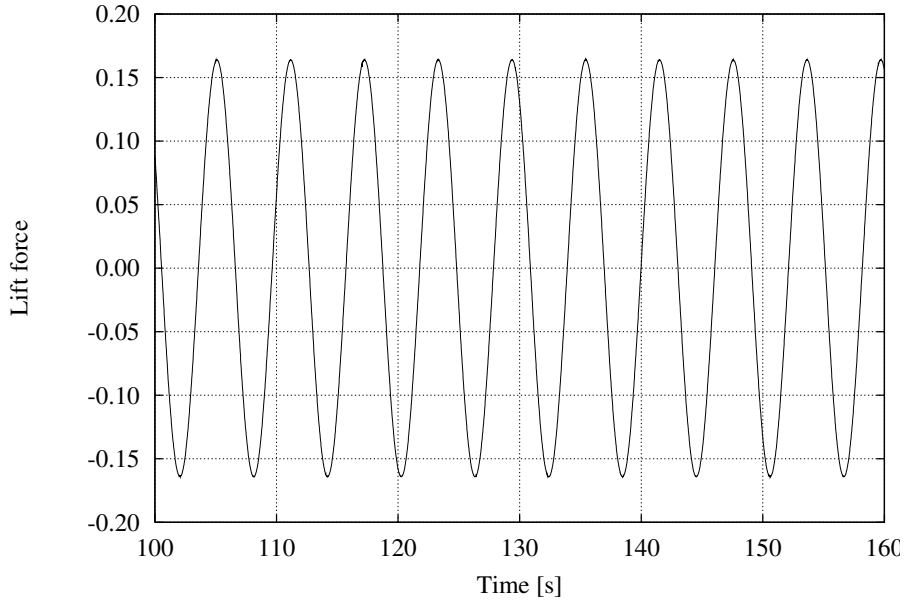


Figure 14. 2D flow around a cylinder. Time evolution of lift force

8.4 2D viscous flow around a Naca airfoil

The viscous flow around a NACA 0012 airfoil for an angle of attack of zero degrees and $Re = 10000$ was analyzed. Laminar flow conditions were again assumed.

Figure 15 shows the geometry of the domain and the boundary conditions. The grid of 14249 points chosen is shown on Figure 16. A finer layer of 972 points was used around the airfoil to capture viscous effects as shown in the figure.

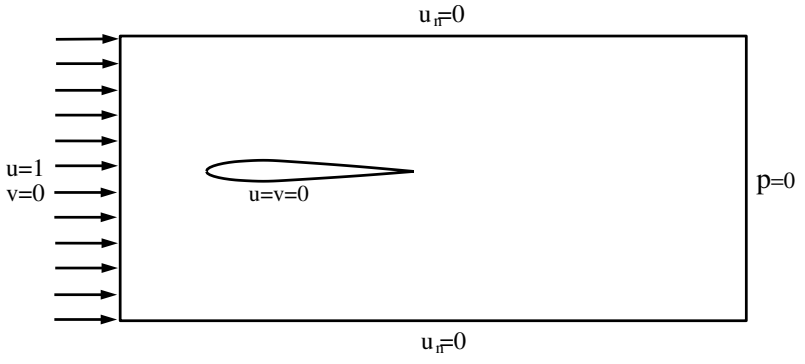


Figure 15. 2D flow around a NACA airfoil. $\alpha = 0°$, $Re = 10000$. Analysis domain and boundary conditions

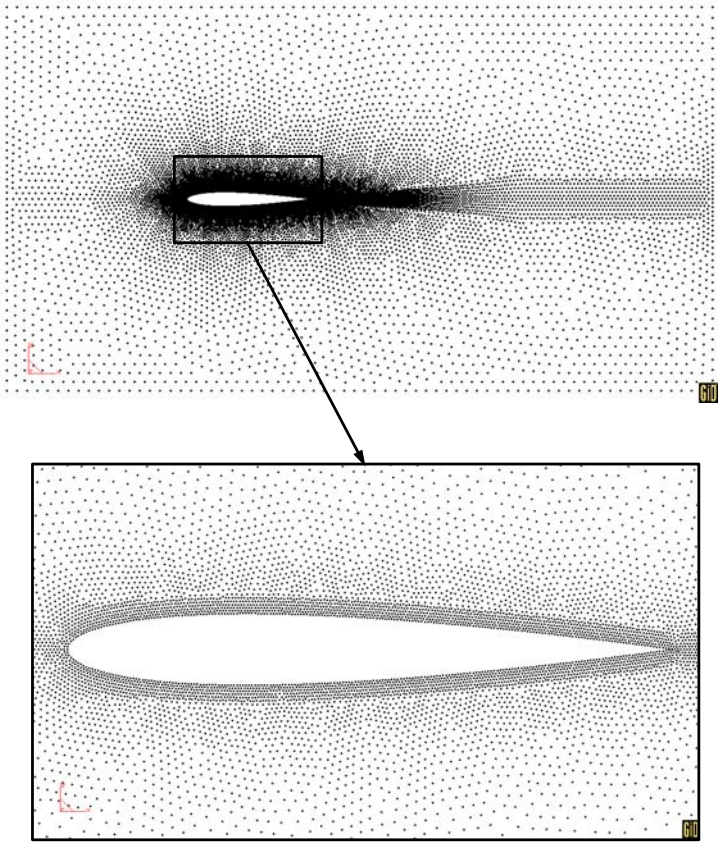
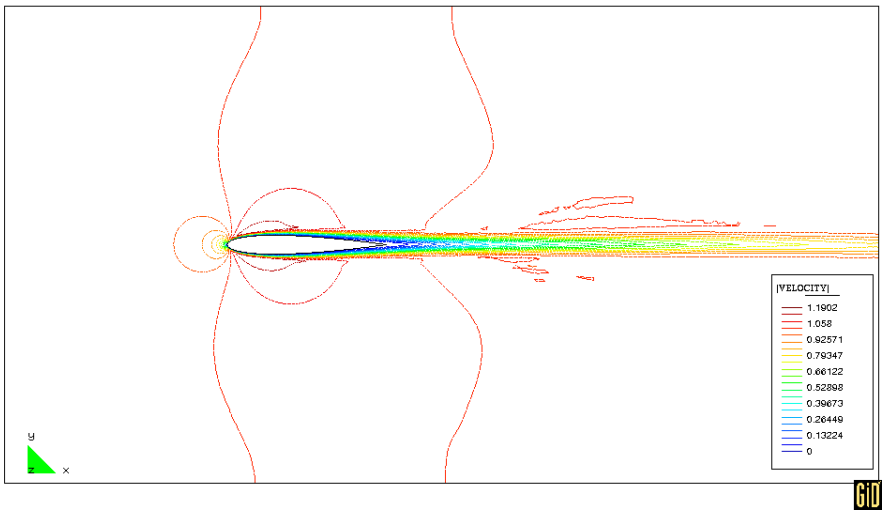


Figure 16. Distribution of 14249 points for analysis of a NACA airfoil. Detail of boundary layer of 972 point to capture viscous effects

Figure 17 shows some numerical results of the velocity streamlines for the steady state situation. Note the well developed wake at the back of the airfoil. A close up of the streamlines next to the airfoil showing the boundary layer developed is also presented.

a)



b)

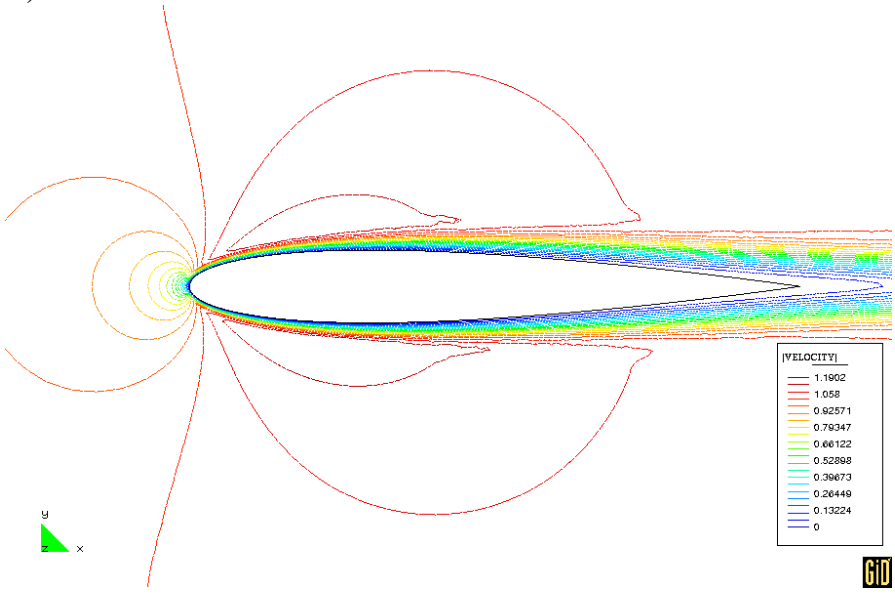


Figure 17. 2D analysis of a NACA airfoil. Velocity streamlines at steady state for $\alpha = 0^\circ$ and $Re = 10000$

9 FINAL CONCLUSSIONS

The stabilized equations for a viscous incompressible fluid using the finite calculus procedure are the basis for deriving a stabilized finite point method for the meshless solution of incompressible flows. The three step semi-implicit fractional

scheme provides a simple and accurate procedure for both transient and steady state solutions using equal order interpolation for the velocities and the pressure. The stabilized FPM is a promising technique for the practical meshless solution of industrial flow problems.

REFERENCES

[1] J.J. Monaghan, "Smoothed particle hydrodynamics: Some recent improvement and applications", *Annu. Rev. Astron. Physics*, **30**, p. 543, (1992).

[2] P.W. Randles and L.D. Libersky, "Smoothed particle hydrodynamics: Some recent improvement and applications", *Appl. Mech. Engng.*, **139**, p. 175, (1996).

[3] N. Perrone and R. Kao, *A general finite difference method for arbitrary meshes.*, *Comp. Struct*, **5**, pp. 45-47, (1975).

[4] T. Liszka and J. Orkisz, "The finite difference method at arbitrary irregular grids and its application in applied mechanics", *Comp. Struct.*, **11**, pp. 83-95, (1980).

[5] B. Nayroles, G. Touzot and P. Villon, "Generalizing the FEM: Diffuse approximation and diffuse elements. Comput. Mechanics", **10**, pp. 307-318, (1992).

[6] T. Belytschko, Y. Lu and L. Gu, "Element free Galerkin methods.", *Int. J. Num. Meth. Engng.*, **37**, pp. 229-56, (1994).

[7] J. Dolbow and T. Belytschko, "An introduction to programming the meshless element free Galerkin method", *Archives of Comput. Meth. in Engng.*, **5**, (3), pp. 207–241, (1998).

[8] W.K. Liu, S. Jun, S. LI, J. Adee and T. Belytschko, "Reproducing Kernel particle methods for structural dynamics", *Int. J. Num. Meth. Engng.*, **38**, pp. 1655–1679, (1995).

[9] W.K. Liu, Y Chen, S. Jun, J.S. Chen, T. Belytschko, C. Pan, R.A. Uras and C.T. Chang, "Overview and applications of the Reproducing Kernel particle method", *Archives of Comput. Meth. in Engng.*, Vol. **3**, (1), pp. 3–80, (1996).

[10] E. Oñate, S. Idelsohn, O.C. Zienkiewicz and T. Fisher, "A finite point method for analysis of fluid flow problems", *Proceedings of the 9th Int. Conference on Finite Element Methods in Fluids*, Venize, Italy, pp. 15-21, October, (1995).

[11] E. Oñate, S. Idelsohn, O.C. Zienkiewicz and R.L. Taylor, "A finite point method in computational mechanics. Applications to convective transport and fluid flow", *Int. J. Num. Meth. Engng.*, Vol. **39**, pp. 3839–3866, (1996).

[12] E. Oñate, S. Idelsohn, O.C. Zienkiewicz and R.L. Taylor, "A stabilized finite point method for analysis of fluid mechanics's problems", *Comput. Meth. in Appl. Engng.*, Vol. **139**, pp. 1-4, pp. 315–347, (1996).

[13] E. Oñate and S. Idelsohn, "A mesh free finite point method for advective-diffusive transport and fluid flow problems", *Computational Mechanics*, **21**, pp. 283–292, (1998).

[14] E. Oñate, C. Sacco and S. Idelsohn, "A finite point method for incompressible flow problems", *Computing and Visualization in Sciences*, **3**, 67–75, 2000.

[15] R. Löhner, C. Sacco, E. Oñate and S. Idelsohn, "A finite point method for compressible flow", to be published in *Int. J. Num. Meth. Engng.*, 2000.

[16] E. Oñate, "Derivation of stabilized equations for advective-diffusive transport and fluid flow problems", *Comput. Meth. Appl. Mech. Engng.*, Vol. 151, 1-2, pp. 233–267, (1998).

[17] E. Oñate, J. García and S. Idelsohn, "Computation of the stabilization parameter for the finite element solution of advective-diffusive problems", *Int. J. Num. Meth. Fluids*, Vol. 25, pp. 1385–1407, (1997).

[18] E. Oñate, J. García and S. Idelsohn, "An alpha-adaptive approach for stabilized finite element solution of advective-diffusive problems with sharp gradients", *New Adv. in Adaptive Comp. Met. in Mech.*, P. Ladeveze and J.T. Oden (Eds.), Elsevier, (1998).

[19] E. Oñate, "A finite element method for incompressible viscous flows using a finite increment calculus formulation", *Comput. Meth. Appl. Mech. Engng.*, **182**, 355–370, 2000.

[20] E. Oñate and M. Manzán, "A general procedure for deriving stabilized space-time finite elements for advective-diffussive problems", *Int. J. Num. Meth. in Fluids*, **31**, 203–221, 1999.

[21] E. Oñate and J. García, "A finite element method for fluid-structure interaction with surface waves using a finite calculus formulation", Submitted to *Comp. Meth. Appl. Mech. Engng.*

[22] O.C. Zienkiewicz and R.L. Taylor, "*The finite element method.*", 5th Edition, Arnold, (2000).

[23] A.Brooks and T.J.R. Hughes, "Streamline upwind/Petrov-Galerkin formulation for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations", *Comput. Meth. Appl. Mech. Engng*, **32**, 199–259, 1982.

[24] T.J.R. Hughes and M. Mallet, "A new finite element formulations for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective-diffusive systems", *Comp. Meth. Appl. Mech. Engng.*, **58**, pp. 305–328, 1986.

[25] P. Hansbo and a. Szepessy, "A velocity-pressure streamline diffusion finite element method for the incompressible Navier-Stokes equations", *Comp. Meth. Appl. Mech. Engng.*, **84**, 175–192, 1990.

[26] T.J.R. Hughes, L.P. Franca and M. Balestra, "A new finite element formulation for computational fluid dynamics. V Circumventing the Babuska-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accomodating equal order interpolations", *Comp. Meth. Appl. Mech. Engng.*, **59**, 85–89, 1986.

[27] L.P. Franca and S.L. Frey, "Stabilized finite element methods: II. The incompressible Navier-Stokes equations", *Comput. Meth. Appl. Mech. Engn*, Vol. **99**, pp. 209–233, 1992.

[28] T.J.R. Hughes, G. Hauke and K. Jansen, "Stabilized finite element methods in fluids: Inspirations, origins, status and recent developments", in: *Recent Developments in Finite Element Analysis*. A Book Dedicated to Robert L. Taylor, T.J.R. Hughes, E. Oñate and O.C. Zienkiewicz (Eds.), (International Center for Numerical Methods in Engineering, Barcelona, Spain, pp. 272–292, 1994.

[29] M.A. Cruchaga and E. Oñate, "A finite element formulation for incompressible flow problems using a generalized streamline operator", *Computer Methods in Applied Mechanics and Engineering*, 143, 49–67, 1997.

[30] M.A. Cruchaga and E. Oñate, "A generalized streamline finite element approach for the analysis of incompressible flow problems including moving surfaces", *Computer Methods in Applied Mechanics and Engineering*, 173, 241–255, 1999.

[31] T.J.R. Hughes, L.P. Franca and G.M. Hulbert, "A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective-diffusive equations", *Comput. Meth. Appl. Mech. Engng.*, **73**, pp. 173–189, 1989.

[32] T.E. Tezduyar, S. Mittal, S.E. Ray and R. Shih, "Incompressible flow computations with stabilized bilinear and linear equal order interpolation velocity–pressure elements", *Comp. Meth. Appl. Mech. Engng.*, **95**, 221–242, 1992.

[33] O.C. Zienkiewicz and R. Codina, "A general algorithm for compressible and incompressible flow. Part I: The split characteristic based scheme", *Int. J. Num. Meth. in Fluids*, **20**, 869-85, (1995).

[34] O.C. Zienkiewicz, K. Morgan, B.V.K. Satya Sai, R. Codina and M. Vázquez, "A general algorithm for compressible and incompressible flow. Part II: Tests on the explicit form", *Int. J. Num. Meth. in Fluids*, **20**, No. 8-9, 886-913, 1995.

[35] T.J.R. Hughes, "Multiscale phenomena: Greens functions, subgrid scale models, bubbles and the origins of stabilized methods", *Comput. Meth. Appl. Mech. Engng*, Vol. **127**, pp. 387–401, 1995.

[36] R. Codina, "A stabilized finite element method for generalized stationary incompressible flows", Publication PI-148, CIMNE, Barcelona, February 1999.

[37] R. Codina and J. Blasco, "Stabilized finite element method for the transient Navier-Stokes equations based on a pressure gradient operator". To appear in *Computer Methods in Appl. Mech. Engng.*

[38] F. Brezzi, M.O. Bristeau, L.P. Franca, M. Mallet and G. Rogé, "A relationship between stabilized finite element methods and the Galerkin method with bubble functions", *Comput. Meth. Appl. Mech. Engn.*, Vol. **96**, pp. 117–129, 1992.

[39] F. Brezzi, D. Marini and A. Russo, "Pseudo residual-free bubbles and stabilized methods", *Computational Methods in Applied Sciences '96*, J. Periaux *et. al.* (Eds.), J. Wiley, 1996.

[40] F. Brezzi, L.P. Franca, T.J.R. Hughes and A. Russo, "$b = \int g$", *Comput. Meth. Appl. Mech. Engn.*, **145**, 329–339, 1997.

[41] T.J.R. Hughes and M. Mallet, "A new finite element formulations for computational fluid dynamics: IV. A discontinuity capturing operator for multidimensional advective-diffusive system, *Comput. Meth. Appl. Mech. Engng.*, **58**, 329–336, 1986.

[42] A.C. Galeão and E.G. Dutra do Carmo, "A consistent approximate upwind Petrov-Galerkin method for convection dominated problems", *Comput. Meth. Appl. Mech. Engng.*, **68**, 83–95, 1988.

[43] R. Codina, "A discontinuity-capturing crosswind dissipation for the finite element solution of the convection-diffusion equation", *Comput. Meth. Appl.*

Mech. Engng., **110**, 325–342, 1993.

[44] R. Codina, "Comparison of some finite element methods for solving the diffusion-convection-reaction equation". *Comp. Meth. Appl. Mech. Engng.*, **188**, 61–82, 2000.

[45] R. Codina, "On stabilized finite element methods for linear systems of convection-diffusion-reaction equation", Publication CIMNE PI-162, December 1997. To appear in *Computer Meth. Appl. Mech. Engng.*

[46] U. Ghia, K. Ghia and C. Shin, "High-*Re* solutions for incompressible flow using the Navier-Stokes equation and a multi-grid method", *J. Comp. Phys.*, Vol. **48**, pp. 387–441, (1982).

[47] B.F. Armaly, F. Durst, J.C.F. Pereira and B. Schönung, "Experimental and theoretical investigations of backword-facing step flow", *J. Fluid Mech.*, Vol. **127**, pp. 473–496, (1983).

[48] A. Roshko, "On the development of turbulent wakes form vortex streets", NACA Technical Report No. 1191, National Advisory Committee for Aeronautics, USA, (1954).